IN THE CLAIMS

1. (Original) A method of translating a virtual address to a physical address in a real time operating system, the method comprising:

indexing into a first level table using a portion of the virtual address;

generating an offset to a second level table based on an entry in the first level table combined with a portion of the virtual address; and

combining the virtual address with an entry in the second level table obtained using the offset.

2. (Original) The method of claim 1 wherein the entry in the second level table comprises multiple control bits for all pages of a decode area and valid bits for each page in the decode area.

3. (Original) The method of claim 1 wherein a single half-word in the second level table corresponds to two hardware register words.

4. (Original) The method of claim 3 wherein the single half-word is formed to minimize operations of a computer implementing the method.

5. (Original) A method of translating a virtual address to a physical address in a real time operating system, the method comprising:

indexing into a first level table using a portion of the virtual address;

generating an offset to a second level table based on an entry in the first level table combined with a portion of the virtual address;

reading a second level table entry pointed to by the offset; and

concatenating the virtual address with bits from the entry in the second level table and a process ID to fill control hardware registers.

6. (Original) A computer readable medium having instructions for causing a computer to perform a method of translating a virtual address to a physical address in a real time operating system, the method comprising:

indexing into a first level table using a portion of the virtual address;

generating an offset to a second level table based on an entry in the first level table combined with a portion of the virtual address;

reading a second level table entry pointed to by the offset; and

concatenating the virtual address with bits from the entry in the second level table and a process ID to fill control hardware registers.

7. (Original) A method of translating a virtual address to a physical address in a real time operating system, the method comprising:

indexing into a first level table using a portion of the virtual address;

generating an offset to a second level table based on an entry in the first level table combined with a portion of the virtual address;

reading a second level table entry pointed to by the offset;

indexing into a valid bit array of the second level table entry to identify a valid bit;

if the valid bit is valid, concatenating the virtual address with bits from the entry in the second level table and a process ID to fill control hardware registers; otherwise,

forming an index into a level three table from the level two entry and selected bits of the virtual address; and

concatenating the virtual address with bits from the entry in the third level table and a process ID to fill control hardware registers.

8. (Original) The method of claim 7 wherein a single half-word in the second level table corresponds to two hardware register words and wherein a single word in the level three table corresponds to two hardware register words.

9. (Original) A method of translating a virtual address to a physical address of a memory for a computer system running a real time operating system, the method comprising:

using a first translation algorithm having at least one level of indirection for a first partition of memory; and

using a second translation algorithm having no indirection for a second partition of memory.

10. (Original) The method of claim 9 wherein the first translation algorithm uses a series of three translation lookaside buffer tables.

11. (Original) The method of claim 9 wherein the algorithms generate control words to translate the address for a majority of blocks of memory.

12. (Original) A system for translating a virtual address to a physical address in a real time operating system, the system comprising:

means for indexing into a first level table using a portion of the virtual address;

means for generating an offset to a second level table based on an entry in the first level table combined with a portion of the virtual address; and

means for combining the virtual address with an entry in the second level table obtained using the offset.

13. (Original) A system for translating a virtual address to a physical address in a real time operating system, the system comprising:

a module that indexes into a first level table using a portion of the virtual address;

a module that generates an offset to a second level table based on an entry in the first level table combined with a portion of the virtual address;

a module that reads a second level table entry pointed to by the offset; and

a module that concatenates the virtual address with bits from the entry in the second level table and a process ID to fill control hardware registers.

14. (Original) A system for translating a virtual address to a physical address in a real time operating system, the system comprising:

a first translation mechanism having at least one level of indirection for a first partition of memory; and

a second translation mechanism having no indirection for a second partition of memory.

15. (Original) The system of claim 14 wherein the first translation mechanism comprises a series of three translation lookaside buffer tables.

16. (Original) The system of claim 14 wherein the translation mechanisms comprises tables of control words that allow translation of the address for a majority of blocks of memory.

17. (Original) A system for translating a virtual address to a physical address in a real time operating system, the system comprising:

a translator;

a first table indexed by a portion of the virtual address, the first table having an entry comprising an offset; and

a second table indexed by a combination of the offset from the first table entry and a portion of the virtual address, the second table having a control word that allows translation of the virtual address to a physical address.

18. (Original) The system of claim 17 and further comprising:

an entry in the second table containing an offset; and

a third table indexed by a combination of the offset from the second table entry and a portion of the virtual address, the third table having a control word that allows translation of the virtual address to a physical address.

19. (Original) The system of claim 17 wherein the second table comprises a valid array, and wherein a valid bit within the array is accessed by a portion of the virtual address.

20. (Original) A method of translating a virtual address to a physical address in a real time operating system, the method comprising:

indexing into a first level table using a portion of the virtual address and a base address register or location;

generating an offset to a second level table based on an entry in the first level table combined with a portion of the virtual address and using the level 2 offset register or location field added to the base address; and

combining the virtual address with a process identifier register field and with an entry in the second level table obtained using the offset.

21. (Original) The method of claim 20 wherein the entry in the second level table comprises multiple control bits for all pages of a decode area and valid bits for each page in the decode area.

22. (Original) The method of claim 20 wherein a single half-word in the second level table corresponds to two hardware register words for all memory pages in the block.

23. (Original) The method of claim 20 wherein multiple user programs in partitions co-exist, with each partition having a unique set of tables which are selected by the operating system upon partition activation by loading the base address register or location for the table corresponding to that partition.

24. (Original) The method of claim 23 wherein one or more user processes in a single partitions co-exist, with each process using the loaded base dispatch table and having a unique set of lower

level tables which are selected by the operating system upon partition activation by loading the level 2 offset register or location for the table corresponding to that process.